

# Optimisation entropique des systèmes de détection distribuée parallèle

## Entropy based optimisation for the distributed detection problem

par Denis POMORSKI, Christophe DESROUSSEAUX

Laboratoire d'Automatique et d'Informatique Industrielle de Lille, UPRES A 8021,  
Bâtiment P2, Cité Scientifique, Université Lille I  
59655 Villeneuve d'Ascq cedex, France  
e-mail : [denis.pomorski@univlille1.fr](mailto:denis.pomorski@univlille1.fr)

### *résumé et mots clés*

L'optimisation d'un système de détection distribuée parallèle comprenant  $N$  capteurs aboutit toujours à un système de  $2^N + N$  équations non linéaires couplées, qui n'est résolu pour l'instant que pour des cas particuliers (en supposant par exemple l'indépendance des observations locales) et pour des systèmes comportant peu de capteurs. Le nombre d'équations à résoudre simultanément augmente très rapidement avec le nombre de capteurs. Les calculs nécessaires à la résolution de ces équations deviennent alors très vite inextricables.

Dans cette contribution, une procédure de sélection de capteurs pertinents pour le processus de décision basée sur l'utilisation de l'entropie conditionnelle de Shannon est développée. Puis, ces systèmes sont optimisés via une méthode entropique. Celle-ci détermine les seuils locaux et construit un arbre de décision (qui représente l'opérateur de fusion) permettant de minimiser la probabilité d'erreur de décision. Les intérêts de cette approche sont de pallier le problème du choix de l'opérateur de fusion, d'être très simple à implémenter, et d'être très peu coûteuse en termes de calculs.

Les performances des systèmes distribués parallèles étant moins bonnes que celles des systèmes centralisés, les techniques d'optimisation précédentes seront étendues au problème de la quantification répartie afin d'obtenir un compromis entre la quantité d'information à envoyer à l'opérateur de fusion et les performances souhaitées du système de décision.

détection distribuée parallèle, entropie de Shannon, arbre de décision, quantification répartie.

### *abstract and key words*

The optimisation of a parallel distributed detection system with  $N$  sensors always leads to a set of  $2^N + N$  non linear equations, which is only solved in particular cases, assuming statistical independence of the local observations and for systems constituted by very few sensors. Usually, the number of equations to solve increases very quickly with the number of sensors. The computations become unfeasible.

In this paper, a selection procedure of sensors relevant for the decision process based on the use of Shannon's conditional entropy is developed. Then, these systems are optimized via an entropy based method. This one determines the local thresholds and constructs a decision tree which minimises the decision error probability. This approach allows to avoid the fusion operator choice problem. Moreover, this approach is very simple and operational in computations terms.

Due to the fact that the performances of distributed systems are lower than the centralised ones, the previous optimisation techniques can be applied in the distributed quantification problem taking into account a compromise between the information flow to broadcast to the fusion center, and the performances of the decision system.

distributed detection, Shannon's entropy, decision tree, distributed quantification.

# 1. introduction

Considérons  $Y = (Y_1, Y_2, \dots, Y_N)$  un vecteur de capteurs et  $y = (y_1, y_2, \dots, y_N)$  le vecteur des mesures. À partir de  $y$ , le système de détection doit prendre une décision binaire  $u_0 \in \{0, 1\}$  ( $u_0 = i$  si  $H_i$  est décidée).

Les probabilités *a priori*  $P_i$  ( $i = 0, 1$ ) et les densités de probabilités conditionnelles  $p(y/H_i)$  ( $i = 0, 1$ ) sous chaque hypothèse sont supposées connues, et on note :

- $P_D = P(u_0 = 1/H_1)$  la probabilité de détection,
- $P_F = P(u_0 = 1/H_0)$  la probabilité de fausse alarme.

Nous ne traitons pas du problème de la gestion de ressources (notamment de la localisation des grandeurs à mesurer) en général liée au problème de poursuite, ou *tracking* [1] [2] et pour lequel l'aspect temporel doit être introduit. Dans ce contexte, les phases d'expertise et de fonctionnement opérationnel doivent être distinctes. Il est alors nécessaire de prendre en compte les deux points suivants :

- un caractère d'urgence (par exemple) nécessite l'introduction d'une étape d'anticipation (les calculs ne sont pas finis, et il faut décider de suite) qui consiste à proposer une décision à un instant donné, avec un degré de fiabilité. Cette notion rejoint le principe des algorithmes « *anytime* » ;
- dans certains cas, il est nécessaire de ne pas proposer une décision peu fiable. Il est alors utile d'attendre quelques informations supplémentaires afin de conforter notre décision (*SPRT* : *Sequential Probability Ratio Test*) [3]

Notre objectif est d'optimiser le système de détection à partir de la seule connaissance statistique que l'on a sur celui-ci, et nous ne prenons pas en compte les critères relatifs à la nature d'un système particulier, pour lequel les phases de modélisation ou d'identification sont très importantes. La prise de décision consiste alors à poser un problème d'optimisation, qui peut faire intervenir différents critères :

- minimisation du nombre total de bits à envoyer à l'opérateur de fusion,
- critère de qualité de la détection, fonction des probabilités de fausse alarme et de détection (voir 3.1.), ...

Ce problème multicritère peut être résolu en définissant un critère composite, fonction des critères de base (figure 1).

Il est alors nécessaire de définir les « poids » respectifs de chacun des critères pris en compte (les unités sont différentes). Nous nous affranchissons de cette connaissance subjective en fixant *a priori* l'un des deux critères, et en cherchant la solution qui permettrait d'optimiser le deuxième (figure 2) [4].

Dans la littérature, l'architecture retenue est souvent une contrainte, et l'objectif est alors de maximiser la qualité de la décision. La détection d'un signal à partir d'observations issues de plusieurs capteurs est envisagée de deux manières différentes.

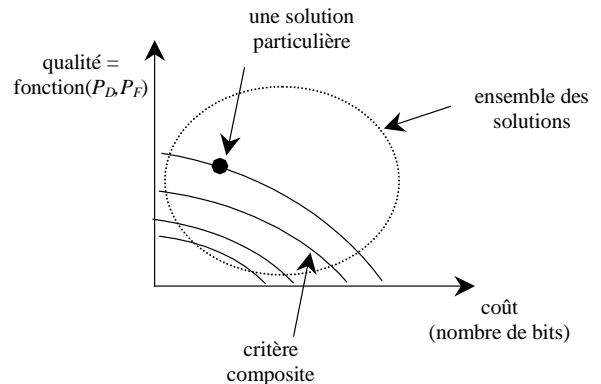


Figure 1. – Critère composite.

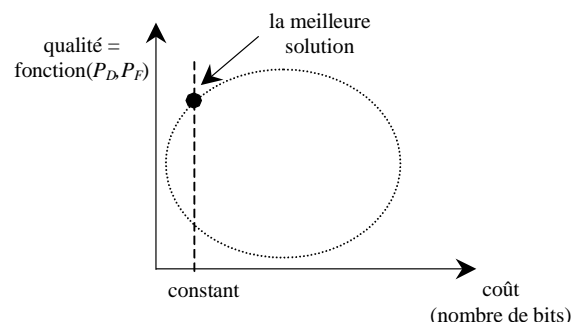


Figure 2. – Proper Equality Constraint.

Traditionnellement, l'ensemble des capteurs communiquent leurs observations directement au détecteur central qui prend la décision. Cette approche, appelée *détection centralisée*, nécessite souvent des lignes de communication à large bande si l'on veut obtenir une décision en temps réel.

La deuxième approche, appelée *détection distribuée*, consiste à associer à chaque capteur un détecteur qui décide localement si un signal a été détecté ou non. Notons  $u_i$  la sortie du détecteur  $i$ . Deux architectures sont souvent étudiées :

- l'architecture distribuée parallèle [5] [6] (figure 3). Les décisions locales sont envoyées à un opérateur de fusion qui les combine afin de prendre la décision finale.
- et l'architecture distribuée série [7] [8] [9] [10] (figure 4), pour laquelle chaque détecteur local reçoit une information issue d'un capteur et envoie un message binaire à son successeur. La décision du premier détecteur est basée sur l'information issue d'un capteur uniquement, et le dernier détecteur élabore la décision finale.

Dans la littérature, les critères de Bayes et de Neyman-Pearson [11] sont très utilisés afin d'optimiser les systèmes de détection centralisée et distribuée. Dans l'approche Bayésienne, un coût est associé à chaque situation et une fonction risque moyen est minimisée. Dans les applications pour lesquelles ces coûts sont connus et ont une signification précise, cette démarche donne d'excellents résultats. L'approche de Neyman-Pearson consiste, quant à elle, à chercher la meilleure probabilité de détection pour une probabilité de fausse alarme donnée.

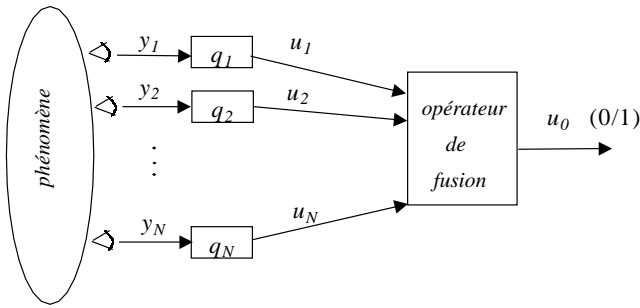


Figure 3. – Architecture d'un système de détection distribuée parallèle.

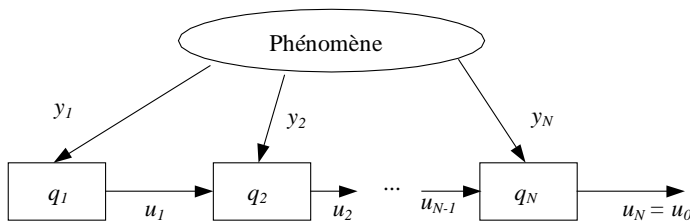


Figure 4. – Architecture d'un système de détection distribuée série

Pourtant, dans certaines applications (en particulier lorsque les probabilités des hypothèses *a priori* sont mal connues, ou lorsque l'une d'elles est proche de zéro), il est intéressant de poser le problème de l'optimisation en considérant la quantité d'information pertinente pour le problème de détection. Cette approche pourrait notamment être très intéressante pour les problèmes de communication numérique où la quantité d'information à envoyer est plus intéressante que l'information elle-même. Pour de tels systèmes, un critère basé sur l'entropie semble être approprié [11].

Enfin, dans le cas de la détection distribuée parallèle, en utilisant les critères de Bayes [5] [12] [6] [13] [14] [10] ou de Neyman-Pearson [15] [16] [17] [18] [19], l'optimisation d'un système comprenant  $N$  capteurs aboutit toujours à un système de  $2^N + N$  équations non linéaires couplées à résoudre, correspondant à l'optimisation de l'opérateur de fusion et des seuils des détecteurs locaux.

Dans le cas de la détection distribuée série, on aboutit à un système de  $2N - 1$  équations non linéaires couplées à résoudre, correspondant à l'optimisation des seuils des détecteurs locaux.

Ces systèmes n'ont pour l'instant pu être résolus que pour des cas particuliers en supposant par exemple l'indépendance des observations locales et pour des systèmes comportant peu de capteurs.

Ce travail traite de l'optimisation des architectures distribuées parallèles. Considérons par exemple le contexte de la surveillance d'installations industrielles complexes, où un grand nombre de capteurs (délivrant une information numérique et/ou symbolique) observant des grandeurs physiques différentes est

disponible. L'optimisation d'un système de détection distribuée utilisant tous les capteurs s'avère très vite irréalisable. Dans certains cas, ceux-ci peuvent en effet présenter des redondances, sans améliorer les performances de l'ensemble, ou noyer le système sous un flot d'informations trop coûteux à gérer. C'est pour cette raison que nous proposons d'introduire dans les systèmes de détection distribuée parallèle, avant toute optimisation, une étape de sélection de capteurs [20]. Parmi tous les capteurs disponibles, nous proposons de ne faire intervenir que ceux apportant une information significative au processus de décision.

La sélection de capteurs pertinents au processus de décision étant faite, on pourra optimiser le système en utilisant les résultats classiques de la théorie de la détection [11], ou en utilisant une démarche informationnelle. Dans ce cadre, Middleton [21] et Gabrielle [22] ont proposé une architecture de détection centralisée basée sur la minimisation de la perte d'information entre l'entrée et la sortie du système. Hoballah [14] [23], et Warren [24] ont étendu ce résultat au problème de la détection distribuée parallèle. Nous proposons de reformuler le problème afin d'aboutir à un formalisme proche de celui obtenu en utilisant les critères de Bayes et de Neyman-Pearson. C'est ainsi que l'entropie conditionnelle  $h(H/u_0)$  (où  $H$  est l'hypothèse vraie et  $u_0$  est la décision prise) [23] [24] [25] représentant l'incertitude sur  $H$  connaissant la décision  $u_0$  est minimisée. Or, le problème de la méconnaissance de l'opérateur de fusion conduit en général à tester toutes les combinaisons possibles des décisions locales afin de ne retenir que la combinaison optimale.

Nous proposons de nous affranchir de cette contrainte en développant une démarche utilisant un arbre de décision [26] [27] en guise d'opérateur de fusion. Les seuils associés aux détecteurs locaux<sup>1</sup> sont optimisés, ainsi que la loi de combinaison des décisions locales. Ces algorithmes permettront d'approcher le système de détection distribuée parallèle optimal en limitant la complexité des calculs à mettre en œuvre.

Enfin, les performances des systèmes de détection centralisée étant meilleures que celles des systèmes de détection distribuée, les techniques d'optimisation précédentes seront étendues au problème de la quantification répartie afin d'obtenir un compromis entre la quantité d'information à envoyer à l'opérateur de fusion et les performances du système de détection [4].

## 2. sélection de capteurs pertinents

En détection distribuée parallèle, il est intéressant d'implémenter une structure en ne considérant que le sous-ensemble de cap-

1. L'optimisation d'une architecture série basée sur un critère entropique est développée dans [28]

teurs pertinents pour le processus de décision, plutôt que l'ensemble des informations disponibles. Dans cette optique, l'utilisation d'un critère entropique est proposée.

## 2.1. formulation du problème

L'objectif de la phase de sélection est de déterminer une configuration réduite qui permettra de tirer le meilleur parti de toutes les informations  $y = (y_1, \dots, y_N)$  dont dispose l'utilisateur. Il est par conséquent nécessaire de déterminer le sous-vecteur  $(y_1, \dots, y_p) \in P(y)$  (où  $P(y)$  représente l'ensemble de toutes les variables multidimensionnelles dont les composantes sont des composantes élémentaires de  $y$  et  $p < N$ ) de capteurs tel que l'incertitude relative à  $H$  connaissant l'information fournie par  $(y_1, \dots, y_p)$ , c'est-à-dire  $h(H/(y_1, \dots, y_p))$  (avec  $h(H/(y_1, \dots, y_p)) = E[-\log_2 P(H/(y_1, \dots, y_p))]$ ), se rapproche le plus de la quantité  $h(H/(y_1, \dots, y_N))$ . Un compromis entre les performances et la complexité du système à optimiser est ainsi trouvé.

De plus, les méthodes développées doivent être simples de façon à pouvoir les implémenter sur des systèmes comportant un nombre élevé de capteurs.

Enfin, le nombre de capteurs «  $p$  » à prendre en compte représente bien évidemment la première condition de sortie de l'algorithme. La connaissance de ce nombre n'est pas toujours évidente *a priori*. Une autre démarche consisterait à définir une entropie conditionnelle maximale seuil  $h_{\max}$  en deçà de laquelle la quantité d'information apportée n'aurait plus une grande signification physique. La différence d'entropie conditionnelle  $h_{\max} - h(H/(y_1, \dots, y_N))$  correspondrait en fait à l'erreur qu'on s'autorise en ne prenant en compte que  $p$  capteurs au lieu de  $N$ .

En pratique, l'un des deux critères d'arrêt définis ci-dessus pourra être utilisé, ou les deux conjointement. Si aucune contrainte quant au nombre de capteurs à prendre en compte n'est imposée, il suffit de fixer  $p = N$ . Si  $h_{\max}$  n'est pas utilisé comme critère d'arrêt, il suffit de fixer  $h_{\max} = h(H/(y_1, \dots, y_N))$ .

## 2.2. détection et finesse des variables multidimensionnelles

En détection, un modèle statistique du système étudié semble être une hypothèse presque incontournable, c'est-à-dire que les fonctions de densité de probabilité conditionnelles sous chaque hypothèse  $p(y/H_i)$ ,  $i = 0, 1$  sont connues. Cependant, plus le nombre de capteurs est important, plus il est difficile de disposer d'un modèle statistique du système. Dans ce cas, comme c'est la pratique dans les problèmes de classification supervisée, ces probabilités seront estimées par les fréquences relatives déterminées à partir d'un ensemble d'exemples de taille suffisamment importante.

En détection centralisée, l'utilisation du critère de Bayes ou de Neyman-Pearson aboutit à la même structure optimale du détecteur ; l'ensemble des informations délivrées par les  $N$  capteurs  $Y_i$  ( $i = 1, \dots, N$ ) est transmis à un opérateur de décision qui prend la décision finale  $u_0$  et qui compare, pour chaque observation  $y = (y_1, \dots, y_N)$ , le rapport de vraisemblance à un seuil  $\lambda$ , soit :

$$\Lambda(y) \underset{u_0=0}{\overset{u_0=1}{\gtrless}} \lambda \quad \text{où} \quad \Lambda(y) = \frac{p(y/H_1)}{p(y/H_0)}$$

La décision finale  $u_0$  peut alors être considérée comme une variable moins fine que  $\Lambda(y)$  dans ce sens qu'elle découpe moins finement l'espace des observations.  $\Lambda(y)$  représente de même une variable moins fine que  $y$ . La relation suivante :  $u_0 \succ \Lambda(y) \succ y$  (figure 5) est ainsi obtenue. Le signe «  $\succ$  » représente la relation « être moins fine que... ».

La figure 5 n'illustre que la notion de finesse entre variables. En toute rigueur,  $y$  représente le vecteur des variables mesurées, de dimension  $N$ .

En vertu des propriétés de l'entropie conditionnelle et en tenant compte des relations ci-dessus, les relations suivantes sont obtenues :

$$\begin{aligned} h(H/(y_1, \dots, y_N)) &\leq h(H/\Lambda(y_1, \dots, y_N)) \\ &\leq h(H/u_0) \leq h(H) \leq \log_2 2 = 1^2 \end{aligned}$$

Le même raisonnement peut être repris en détection distribuée parallèle :

$$u_0 \succ (u_1, \dots, u_N) \succ (\Lambda(y_1), \dots, \Lambda(y_N)) \succ (y_1, \dots, y_N)$$

On en déduit de même que :

$$\begin{aligned} h(H/(y_1, \dots, y_N)) &\leq h(H/\Lambda(y_1), \dots, \Lambda(y_N)) \\ &\leq h(H/u_1, \dots, u_N) \leq h(H/u_0) \leq 1 \end{aligned}$$

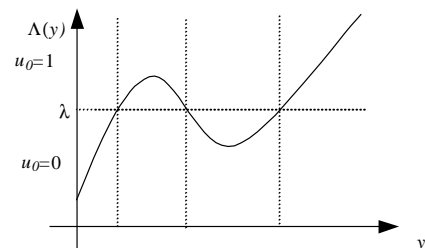


Figure 5. –  $u_0$  est une variable moins fine que  $\Lambda(y)$ , elle-même moins fine que  $y$

2. L'entropie d'une variable possédant  $n$  valeurs est inférieure ou égale à  $\log_2(n)$

### 2.3. quelle méthode utiliser ?

Dans la pratique, le calcul exhaustif de  $h(H/(y_1, \dots, y_p))$  pour tout  $(y_1, \dots, y_p) \in P(y)$  est impossible lorsque le nombre de capteurs est important, et plusieurs heuristiques peuvent être utilisées :

– A partir d'un sous-vecteur quelconque de  $p$  capteurs, une première approche consisterait à utiliser un algorithme de transfert : celui-ci retire le capteur le moins informatif parmi les  $p$  capteurs sélectionnés, puis ajoute le capteur le plus informatif parmi les capteurs restant, et réitère le processus jusqu'à ce que ce vecteur n'évolue plus [29].

– Une démarche hiérarchique ascendante pourrait également être utilisée. Cette approche consisterait à chercher le capteur le plus pertinent (le plus informatif), puis le deuxième, qui, associé au premier, apporte le plus d'information, ..., jusqu'à ce que le (ou les) critère(s) d'arrêt soi(en)t valide(s). Cette approche ne semble pas adaptée à notre problème. En effet, il peut arriver que le pouvoir explicatif de chaque variable prise individuellement d'un vecteur  $S$  soit faible, alors que celui du vecteur  $S$  considéré globalement est important. Dans ce cas, un algorithme prenant en compte les variables une à une, ou ne prenant pas en compte tout le pouvoir explicatif de la variable multidimensionnelle  $S$  globalement, ne permettra pas de mettre en évidence certaines relations existant entre ces variables, et donc ne les choisira pas comme variables pertinentes.

Nous avons choisi d'élaborer une heuristique permettant de rejeter d'emblée les capteurs non informatifs ou redondants au processus de décision en utilisant une démarche hiérarchique descendante.

### 2.4. sélection des capteurs pertinents

Dans un premier temps, le capteur  $y_i$  qui minimise

$$h(H/(y_1, \dots, y_{i-1}, y_{i+1}, \dots, y_N))$$

est écarté de  $y = (y_1, \dots, y_N)$ , ce qui signifie que ce capteur n'apporte que peu d'information sur l'hypothèse vraie par rapport à l'information apportée par  $(y_1, \dots, y_{i-1}, y_{i+1}, \dots, y_N)$ . Le vecteur résiduel des capteurs les plus pertinents est ainsi constitué du vecteur des capteurs  $y$  auquel la composante  $y_i$  a été soustraite.

Dans un deuxième temps, la même démarche est réitérée sur le vecteur résiduel. On écarte alors  $y_j$ , un capteur n'apportant que peu (ou pas) d'information sur l'hypothèse vraie ; et ainsi de suite, ..., jusqu'à :

– ce que le vecteur résiduel soit composé de  $p$  capteurs qui représentent les  $p$  capteurs les plus pertinents pour le problème de la décision,

– ou que l'entropie conditionnelle maximale soit atteinte.

La méthode consiste donc à chercher  $y_i$  ( $y_i \in S_{k-1}$ ) qui minimise  $h(H/(S_{k-1} \setminus y_i))$ , avec  $S_0 = y$  et  $S_k = S_{k-1} \setminus y_i$  ; où  $S_0$

représente le vecteur initial des capteurs pertinents, et  $S_k$ , le vecteur des capteurs conservés au pas  $k$  de l'algorithme (figure 6).

L'algorithme proposé est le suivant :

```

; SELECTION DE CAPTEURS PERTINENTS
; initialisation
0. «  $p$  » et/ou «  $h_{\max}$  » sont des paramètres fixés
1. pour  $i := 1$  à  $N$  ( $N$ =nombre de capteurs potentiels)
2.     Calculer  $h(H/(y \setminus y_i))$ 
3. fin pour
4. Déterminer  $y_i$  qui minimise  $h(H/(y \setminus y_i))$ 
5.  $S \leftarrow y \setminus y_i$ 
; boucle
6. Répéter
7     Calculer  $h(H/(S \setminus y_i))$  avec  $y_i \in S$ 
8.     Déterminer  $y_i$  qui minimise  $h(H/(S \setminus y_i))$ 
9.      $S \leftarrow S \setminus y_i$ 
10. Jusqu'à [ $\text{card} S = p$ ] ou [ $h(H/(S \setminus y_i)) > h_{\max}$ ]
; Fin de l'algorithme

```

avec  $y_i \in S$  et  $S$  appartenant à l'ensemble des variables multidimensionnelles définies sur  $y$ .

**Remarque :** La quantité  $h(H) - h(H/y)$  représentant l'entropie de  $H$  expliquée par  $y$ ,  $q(S/y) = \frac{h(H) - h(H/S)}{h(H) - h(H/y)}$  peut être

définie comme une mesure de la qualité du modèle  $H = g(S)$  par rapport à la qualité du modèle global  $H = f(y)$ . Cette quantité représente la part d'information apportée par  $S$  sur  $H$  par rapport à l'information totale apportée par l'ensemble des capteurs  $y$ , et la maximisation de cette grandeur revient à choisir  $h_{\max}$ ,  $h(H)$  et  $h(H/y)$  étant des constantes.

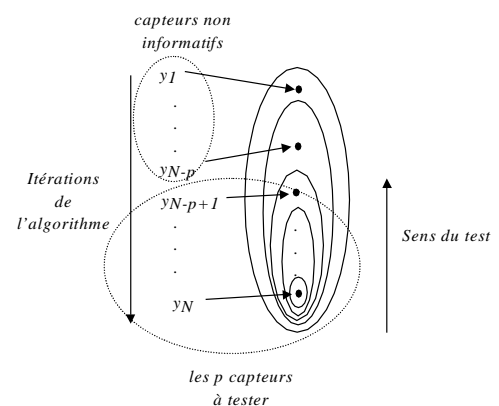


Figure 6. – Sélection des capteurs pertinents, moyennant une renumérotation des capteurs testés. Le premier capteur trouvé par l'algorithme est renommé «  $y_1$  », le deuxième «  $y_2$  », et ainsi de suite...

Afin d'illustrer cette démarche, considérons un système comprenant trois capteurs pour lequel les fonctions de densité de probabilités conditionnelles de chaque détecteur sont des gaussiennes. Sous l'hypothèse  $H_0$ , ces fonctions sont de moyenne 0 et de variance 1. Sous l'hypothèse  $H_1$ , ces fonctions sont de moyennes respectives  $m_1 = 1$ ,  $m_2 = 1.5$  et  $m_3 = 2$  et de variance 1. L'objectif final consiste à sélectionner le couple de capteurs qui permettra la meilleure détection, sans utiliser les trois capteurs disponibles. Remarquons d'emblée qu'intuitivement le couple  $(y_2, y_3)$  semble meilleur que les autres, le chevauchement des gaussiennes étant moindre pour ces deux capteurs. La figure 7 représente les variations de  $h(H)$ ,  $h(H/y_1)$ ,  $h(H/y_2)$ ,  $h(H/y_3)$ ,  $h(H/y_1, y_2)$ ,  $h(H/y_2, y_3)$ ,  $h(H/y_1, y_3)$  et  $h(H/y_1, y_2, y_3)$  en fonction de  $P_0$ .

En utilisant la démarche développée précédemment, le couple  $(y_2, y_3)$  minimise  $h(H/y_i, y_j)$  ( $i, j = 1, 2, 3$ ) et ceci, quelle que soit la valeur prise par  $P_0$ , avec une qualité de modèle minimale de 94 %. Ces résultats sont tout à fait conformes à ce que l'on pouvait attendre par la seule connaissance des densités de probabilité associées aux différents capteurs.

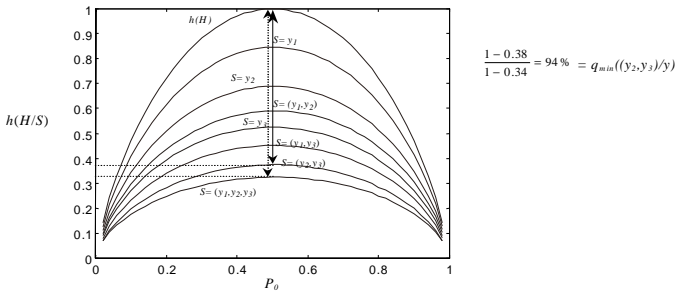


Figure 7. – Variation de  $h(H/S)$  en fonction de  $P_0$ .

### 3. entropie et détection distribuée parallèle

L'optimisation simultanée des détecteurs locaux et de l'opérateur de fusion (figure 3) d'un système à architecture distribuée parallèle, tant du point de vue Bayésien [14] [10] que du point de vue de Neyman-Pearson [15] [19], a été largement étudiée dans la littérature. Une démarche identique est rappelée en utilisant un critère informationnel [11] [28].

#### 3.1. expression de l'entropie $h(H/u_0)$

Notre problématique consiste à optimiser l'architecture de la figure 3 en minimisant l'entropie conditionnelle  $h(H/u_0)$ . Par définition, l'entropie peut se mettre sous la forme suivante :

$$\begin{aligned} h(H/u_0) &= E \left\{ \log \left( \frac{1}{P(H/u_0)} \right) \right\} \\ &= - \sum_{i \in \{-1, 1\}} \left[ \alpha_i \log \frac{\alpha_i}{\alpha_i + \beta_i} + \beta_i \log \frac{\beta_i}{\alpha_i + \beta_i} \right] \end{aligned}$$

avec :

$$\begin{aligned} \alpha_1 &= P_0 P_F & \beta_1 &= (1 - P_0) P_D \\ \alpha_{-1} &= P_0 (1 - P_F) & \text{et} & \beta_{-1} = (1 - P_0) (1 - P_D) \end{aligned}$$

Minimiser  $h(H/u_0)$  revient alors à minimiser une fonction risque moyen pour laquelle les coûts ne sont pas des constantes mais dépendent des probabilités *a posteriori*. L'optimisation du système se fait élément par élément. Dans un premier temps, l'opérateur de fusion est optimisé, les détecteurs locaux étant fixés ; puis chaque détecteur local sera optimisé à son tour.

#### 3.2. optimisation de l'opérateur de fusion

En posant :

$$\begin{aligned} C_{10} &= \log \frac{\alpha_1 + \beta_1}{\alpha_1}, & C_{00} &= \log \frac{\alpha_{-1} + \beta_{-1}}{\alpha_{-1}}, \\ C_{11} &= \log \frac{\alpha_1 + \beta_1}{\beta_1}, & C_{01} &= \log \frac{\alpha_{-1} + \beta_{-1}}{\beta_{-1}} \end{aligned}$$

on a :

$$\begin{aligned} \frac{\partial h(H/u_0)}{\partial v} &= P_0 (C_{10} - C_{00}) \frac{\partial P_F}{\partial v} \\ &\quad - (1 - P_0) (C_{01} - C_{11}) \frac{\partial P_D}{\partial v} \end{aligned}$$

où  $v$  est une variable quelconque.

D'autre part, si le vecteur des décisions prises par les détecteurs locaux est appelé  $u = (u_1, \dots, u_p)$ , nous avons :

$$\begin{aligned} P_D &= P(u_0 = 1/H_1) = \sum_u P(u_0 = 1/u) \cdot P(u/H_1) \\ P_F &= P(u_0 = 1/H_0) = \sum_u P(u_0 = 1/u) \cdot P(u/H_0) \end{aligned}$$

Notons  $v = P(u_0 = 1/u^*)$  la probabilité de décider  $u_0 = 1$  sachant que le vecteur  $u$  est égal à la solution particulière  $u^* = (u_1^*, \dots, u_p^*)$ . L'optimisation élément par élément donne :

$$\frac{\partial h(H/u_0)}{\partial P(u_0 = 1/u^*)} = P_0(C_{10} - C_{00})P(u^*/H_0) - (1 - P_0)(C_{01} - C_{11})P(u^*/H_1)$$

Minimiser  $h(H/u_0)$  revient à fixer  $P(u_0 = 1/u^*) = 1$  lorsque  $h(H/u_0)$  est une fonction décroissante de  $P(u_0 = 1/u^*)$ , et  $P(u_0 = 1/u^*) = 0$  lorsque  $h(H/u_0)$  est une fonction croissante de  $P(u_0 = 1/u^*)$ , ce qui équivaut à appliquer la règle suivante :

$$P_0(C_{10} - C_{00})P(u^*/H_0) - (1 - P_0)(C_{01} - C_{11})P(u^*/H_1) \begin{matrix} \xrightarrow{P(u_0=1/u^*)=0} \\ \geq \\ \xrightarrow{P(u_0=1/u^*)=1} \end{matrix} 0$$

De plus, en faisant l'hypothèse raisonnable que le coût d'une mauvaise décision est supérieur au coût d'une bonne décision, c'est-à-dire que  $C_{10} > C_{00}$  et  $C_{01} > C_{11}$ , la règle de décision précédente peut se mettre sous la forme :

$$\frac{P(u^*/H_1)}{P(u^*/H_0)} \underset{u_0=0}{\overset{u_0=1}{\geq}} \frac{P_0}{1 - P_0} \frac{(C_{10} - C_{00})}{(C_{01} - C_{11})}$$

Cette règle est similaire à celle obtenue en appliquant les critères de Bayes ou de Neyman-Pearson. Les coûts ne sont pas des constantes mais dépendent des probabilités *a posteriori*.

### 3.3. optimisation des détecteurs locaux

L'optimisation élément par élément du détecteur  $k$  ( $k = 1, \dots, p$ ) peut être obtenue en dérivant  $h(H/u_0)$  par rapport à  $P_{F_k}$ . Pour cela, il suffit de poser  $v = P_{F_k}$  :

$$\frac{\partial h(H/u_0)}{\partial P_{F_k}} = P_0(C_{10} - C_{00}) \frac{\partial P_F}{\partial P_{F_k}} - (1 - P_0)(C_{01} - C_{11}) \frac{\partial P_D}{\partial P_{F_k}}$$

Sachant que le seuil  $\lambda_k$  associé au détecteur  $k$  est donné par  $\lambda_k = \frac{\partial P_{D_k}}{\partial P_{F_k}}$ , et que l'on a  $\frac{\partial P_D}{\partial P_{F_k}} = \frac{\partial P_D}{\partial P_{D_k}} \lambda_k$ , on obtient :

$$\frac{\partial h(H/u_0)}{\partial P_{F_k}} = P_0(C_{10} - C_{00}) \frac{\partial P_F}{\partial P_{F_k}} - (1 - P_0)(C_{01} - C_{11}) \frac{\partial P_D}{\partial P_{D_k}} \lambda_k$$

Minimiser  $h(H/u_0)$  revient alors à annuler l'expression ci-dessus, ce qui revient à choisir :

$$\lambda_k = \frac{P_0}{1 - P_0} \frac{(C_{10} - C_{00}) \frac{\partial P_F}{\partial P_{F_k}}}{(C_{01} - C_{11}) \frac{\partial P_D}{\partial P_{D_k}}} \quad (k = 1, \dots, p)$$

Cette structure est identique à celle obtenue en utilisant les critères de Bayes et de Neyman-Pearson. Le calcul des dérivées des équations ci-dessus passe par la prise en compte d'une (ou de plusieurs) fonction(s) de fusion. Nous proposons de nous affranchir de cette contrainte en développant une démarche utilisant un arbre de décision [26] [27] en guise d'opérateur de fusion.

## 4. détection distribuée parallèle par arbre de décision

Dans le cadre de la détection distribuée, que le critère soit le critère de Bayes, de Neyman-Pearson ou que l'on utilise un critère informationnel, la théorie aboutit à la même structure optimale du détecteur. Dans ce contexte, les propriétés de l'entropie peuvent être utilisées afin de limiter la complexité des calculs à mettre en œuvre lors de la détermination de ces seuils.

Dans un premier temps, ces seuils sont déterminés *via* un algorithme hiérarchique ascendant en minimisant l'entropie conditionnelle  $h(H/u_1, \dots, u_p)$ . Cet algorithme permettra d'approcher le système de détection optimal, son principal intérêt étant d'être très peu coûteux en termes de calculs. Etant donné sa simplicité de mise en œuvre, cet algorithme pourra être implémenté sur des systèmes comportant un nombre élevé de capteurs.

Une méthode un peu plus gourmande en termes de calculs est également proposée, à savoir une approche hiérarchique descendante afin de nous affranchir du problème soulevé au paragraphe 2.3.

Une fois connus, ces seuils permettent la construction d'un arbre de décision en minimisant la probabilité d'erreur du système de décision.

### 4.1. détermination des seuils des détecteurs locaux

Cette approche consiste à chercher le couple  $(\lambda_i, y_i)$  qui minimise  $h(H/u_i)$  tel que  $\Lambda(y_i) \underset{u_i=0}{\overset{u_i=1}{\geq}} \lambda_i$ , puis le couple  $(\lambda_j, y_j)$  qui, combiné à  $u_i$ , minimise  $h(H/u_i, u_j)$ , et ainsi de suite, jusqu'à ce que chaque seuil soit déterminé. Il s'agit bien d'une méthode

hiérarchique ascendante, l'évaluation du critère au pas  $k$  de l'algorithme s'effectuant en prenant en compte le choix au pas  $k - 1$  de celui-ci.

La méthode consiste alors à chercher  $\lambda_i$  associé à  $y_i$  ( $y_i \in (y \setminus S_{k-1})$ ,  $\lambda_i \in M_{\Lambda(y_i)}$  et  $\Lambda(y_i) \underset{u_i=0}{\overset{u_i=1}{\geq}} \lambda_i$ ) qui minimise  $h(H/(S_{k-1}, u_i))$ ; avec  $S_0 = \emptyset$ ,  $S_k = (S_{k-1}, u_i)$  et  $M_{\Lambda(y_i)}$  est l'ensemble des valeurs prises par  $\Lambda(y_i)$ . L'algorithme suivant est proposé :

```

; APPROCHE HIERARCHIQUE ASCENDANTE
; DE DETERMINATION DES SEUILS LOCAUX
; initialisation
1. pour  $i := 1$  à  $p$  ( $p = \text{nombre de capteurs pertinents}$ )
2. Calculer les  $h(H/u_i)$  pour tous les  $\lambda_i \in M_{\Lambda(y_i)}$ 
; voir remarque (1)
3. Retenir  $u_i$  et  $\lambda_i$  qui minimisent  $h(H/u_i)$ 
4. fin pour
5.  $S \leftarrow (u_i)$ 
; boucle
6. Répéter
7. Calculer les  $h(H/(S, u_i))$  pour chaque  $\lambda_i \in M_{\Lambda(y_i)}$ 
   avec  $u_i \notin S$ 
; voir remarque (1)
8. Retenir  $u_i$  et  $\lambda_i$  qui minimisent  $h(H/(S, u_i))$ 
9.  $S \leftarrow (S, u_i)$ 
10. Jusqu'à  $[\text{card} S = p]$ 
; Fin de l'algorithme
    
```

Les trois remarques suivantes peuvent être formulées :

- (1) Si le nombre de seuils possibles pour  $\Lambda(y_i)$  est important, la méthode du gradient pourra être utilisée afin de limiter l'aspect combinatoire, ce qui permet de diminuer les temps de calculs.
- (2) Les seuils  $\lambda_i$  étant définis par la méthode proposée, une solution permettant un réajustement de ceux-ci par l'intermédiaire d'une méthode du gradient pourrait être facilement implémentée via une méthode « axe par axe » de la façon suivante : le seuil  $\lambda_1$  est réajusté (les autres  $\lambda_i$  sont constants pour  $i \neq 1$ ) de façon à obtenir un minimum local de l'entropie conditionnelle ; puis l'opération est répétée sur  $\lambda_2, \dots, \lambda_p$  ; et ainsi de suite, jusqu'à ce que les seuils n'évoluent plus.
- (3) Par cette approche ascendante, les seuils obtenus pour chaque détecteur dépendent bien entendu de l'ordre de traite-

ment des capteurs en question. Une démarche de sélection descendante (cf. paragraphe 2.4.) des capteurs peut être envisagée, donnant ainsi l'ordre des capteurs à tester. Il suffira de déterminer les seuils optimaux pour chaque détecteur. L'algorithme suivant est proposé :

```

; APPROCHE HIERARCHIQUE DESCENDANTE
; DE DETERMINATION DES SEUILS LOCAUX
1. Utilisation de l'algorithme de sélection des capteurs
    $\Rightarrow y_p$  est le premier capteur à tester,  $y_{p-1}$ 
   le deuxième, ...,  $y_1$  le dernier
2. Calculer les  $h(H/u_p)$  pour tous les  $\lambda_p \in M_{\Lambda(y_p)}$ 
; voir remarque (1)
3. Retenir  $\lambda_p$  qui minimise  $h(H/u_p)$ 
4.  $S \leftarrow (u_p)$ 
; boucle
5. pour  $i := p - 1$  à 1 faire
6. Calculer les  $h(H/(S, u_i))$  pour chaque  $\lambda_i \in M_{\Lambda(y_i)}$ 
; voir remarque (1)
7. Retenir  $\lambda_i$  qui minimise  $h(H/(S, u_i))$ 
8.  $S \leftarrow (S, u_i)$ 
9. fin pour
; Fin de l'algorithme
    
```

## 4.2. construction de l'arbre de décision

Les seuils ayant été déterminés au moyen de l'un des deux algorithmes précédents, un arbre de décision binaire (figure 8) est construit en guise d'opérateur de fusion. Il permet de minimiser la probabilité d'erreur du système de détection.

Chaque nœud de l'arbre consiste en un seuillage du rapport de vraisemblance d'un capteur  $y_i$  ( $i = 1, \dots, p$ )  $\Lambda(y_i) \underset{u_i=0}{\overset{u_i=1}{\geq}} \lambda_i$  donnant naissance à deux branches, une pour chaque valeur prise par le détecteur local correspondant ( $u_i = 0$  or 1).

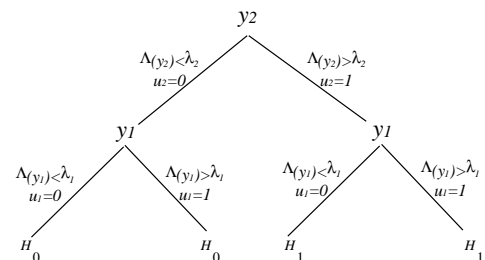


Figure 8. – Arbre de décision trouvé par l'algorithme ascendant sur l'exemple illustratif.



Chaque feuille de l'arbre représente la décision finale  $u_0$  ( $H_0$  ou  $H_1$ ). Cette décision pourra être déterminée en minimisant la probabilité d'erreur, ce qui revient à affecter à chaque feuille une

décision telle que :  $p(\text{chemin et } H_1) \underset{u_0=H_0}{\overset{u_0=H_1}{\geq}} p(\text{chemin et } H_0)$

le chemin étant la concaténation des différents événements élémentaires «  $u_i = k$  » lors du parcours de l'arbre de la racine à la feuille considérée (par exemple  $u_2 = 0$  et  $u_1 = 0$  pour la feuille de gauche de la figure 8).

Illustrons cette démarche en considérant un système formé de deux capteurs dont les observations  $y_1$  et  $y_2$  sont indépendantes. Les fonctions de densité de probabilités sous chaque hypothèse sont des gaussiennes. Sous l'hypothèse  $H_0$ , ces fonctions de densité de probabilités sont de moyenne 0 et de variance 1. Sous l'hypothèse  $H_1$ , ces fonctions sont de moyenne 1 pour  $y_1$  (resp. 1,5 pour  $y_2$ ) et de variance 1.

Les valeurs des seuils  $\lambda_1$  et  $\lambda_2$  optimales (resp. trouvées avec l'algorithme ascendant) sont représentées figure 13 (resp. figure 9) en fonction de  $P_0$ . Les figures 10 et 14 représentent, quant à elles, la variation de  $h(H/u_1, u_2)$  correspondant à ces seuils en fonction de  $P_0$ . Les figures 11 et 12 représentent la variation de la probabilité d'erreur en fonction de  $P_0$ .

Pour  $P_0 = 0.5$ , la première itération de l'algorithme ascendant indique que le capteur  $y_2$  avec le seuil  $\lambda_2 = 1$  minimise l'entropie conditionnelle  $h(H/u_2)$  ; à la deuxième itération, le capteur  $y_1$  avec le seuil  $\lambda_1 = 1$  minimise l'entropie conditionnelle  $h(H/u_1, u_2)$ .

L'arbre de décision permettant de minimiser la probabilité d'erreur du système de détection est représenté figure 8. Sur cet exemple, seule la mesure de  $y_2$  conditionne la décision finale. On retrouve le fait que la fonction de fusion se limite à la décision prise par le capteur 2.

Cet exemple permet de mettre en évidence le fait que les résultats trouvés avec l'algorithme ascendant sont très proches des résultats optimaux trouvés en utilisant le critère entropique, ce qui permet de surcroît, de valider notre approche heuristique qui a pour objectif principal de diminuer l'aspect combinatoire du nombre d'équations non linéaires à résoudre.

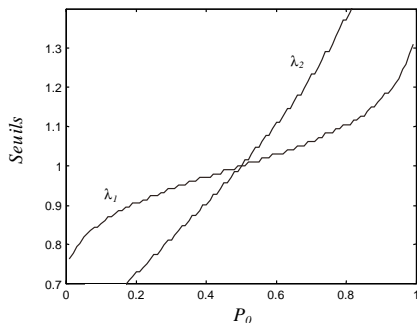


Figure 9. – Seuils trouvés avec l'algorithme ascendant en fonction de  $P_0$ .

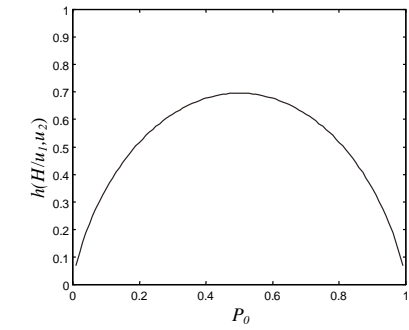


Figure 10. –  $h(H/u_1, u_2)$  correspondant aux seuils trouvés avec l'algorithme ascendant en fonction de  $P_0$ .

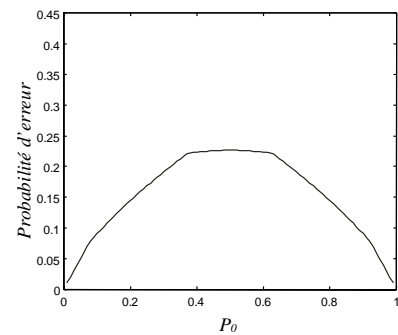


Figure 11. – Probabilité d'erreur en fonction de  $P_0$  correspondant à l'arbre de décision trouvé avec l'algorithme ascendant.

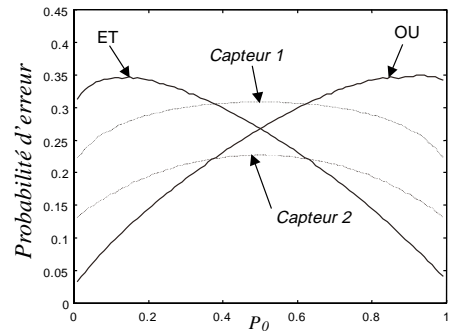


Figure 12. – Probabilité d'erreur en fonction de  $P_0$  correspondant aux seuils optimaux.

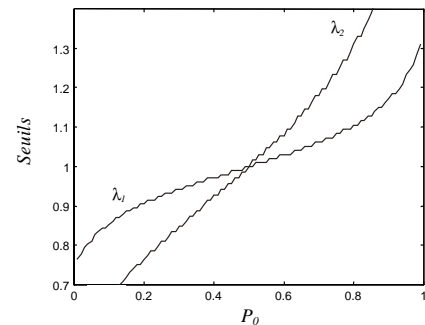


Figure 13. – Seuils optimaux en fonction de  $P_0$ .

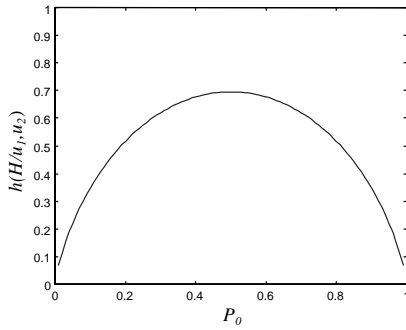


Figure 14. –  $h(H/u_1, u_2)$  correspondant aux seuils optimaux en fonction de  $P_0$ .

## 5. quantification répartie et détection par arbre de décision

La détection centralisée permet de diviser l'espace des observations  $D$  en deux domaines  $D_0$  et  $D_1$  de façon optimale suivant un critère donné (figure 15) car toute l'information utile est envoyée à l'opérateur de fusion qui élabore la décision finale. En détection distribuée parallèle, le rapport de vraisemblance de chaque détecteur est comparé à un seuil. La partition de l'espace des observations qui en résulte ne peut donc être que relativement grossière (figure 16) en comparaison de celle obtenue en détection centralisée.

Il serait pourtant intéressant de trouver un compromis entre les performances optimales mais coûteuses d'un système centralisé, et celles dégradées mais plus économiques des solutions distribuées, l'objectif principal étant de s'approcher au mieux des domaines de décision optimaux trouvés dans le cas centralisé.

Nous proposons d'augmenter le nombre de seuils de chaque détecteur local, de façon à obtenir une partition plus fine de l'espace des observations (figure 17). Ces seuils peuvent être déterminés à partir du critère qui consiste à minimiser  $h(H/u_1, \dots, u_p)$ , pour lequel  $u_1, \dots, u_p$  ne représentent plus des variables moins fines que  $\Lambda(y_1), \dots, \Lambda(y_p)$  en deux classes, mais des variables moins fines en un nombre de classes fixé à l'avance.

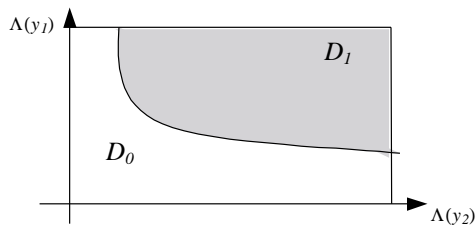


Figure 15. – Les domaines de décision en détection centralisée.

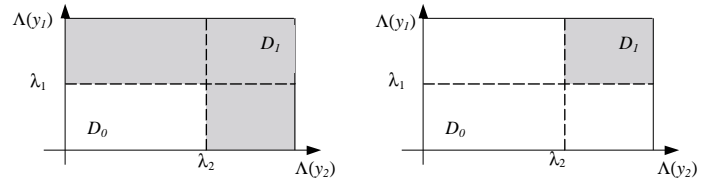


Figure 16. – Les domaines de décision en détection décentralisée parallèle suivant deux règles de fusion (le « ou » et le « et » logique).

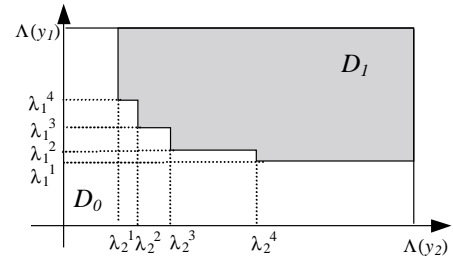


Figure 17. – Les domaines de décision en augmentant le nombre de seuils.

Un algorithme de construction d'arbre de décision du même type que celui développé précédemment peut être repris, en lui ajoutant la possibilité de seuiller le rapport de vraisemblance de chacune des variables suivant plusieurs seuils.

### 5.1. détermination des seuils des détecteurs locaux

La démarche retenue est la suivante :

dans un premier temps, l'algorithme recherche le couple  $(\lambda_i^1, y_i)$  qui minimise  $h(H/u_i^1)$ ,  $u_i^1$  étant une variable moins fine que

$\Lambda(y_i)$  en deux classes telle que  $\Lambda(y_i) \underset{u_i^1=0}{\overset{u_i^1=1}{\geq}} \lambda_i^1$ . Puis, il détermine

le couple  $(\lambda_j^1, y_j)$  qui, associé à  $u_i^1$ , minimise  $h(H/u_i^1, u_j^1)$  ou, s'il existe, le seuil  $\lambda_i^2$  qui, en tenant compte du seuil  $\lambda_i^1$  trouvé précédemment, minimise  $h(H/u_i^2)$ ,  $u_i^2$  représentant une variable moins fine que  $\Lambda(y_i)$  en trois classes, et ainsi de suite, ..., jusqu'à ce que le nombre de seuils «  $nb_{\max}(i)$  » désiré soit déterminé pour chaque détecteur local.

En seuillant au fur et à mesure les rapports de vraisemblance associés aux différents capteurs, une partition de plus en plus fine de l'espace des observations est déterminée (figure 18).

Il suffit alors de définir le nombre maximal autorisé de seuils pour chaque détecteur local ; c'est-à-dire qu'il faut connaître le nombre de bits à envoyer à l'opérateur de fusion. Une autre démarche consisterait à déterminer une entropie conditionnelle maximale seuil  $h_{\max}$  en deçà de laquelle la quantité d'information apportée sur le processus de décision est suffisante.

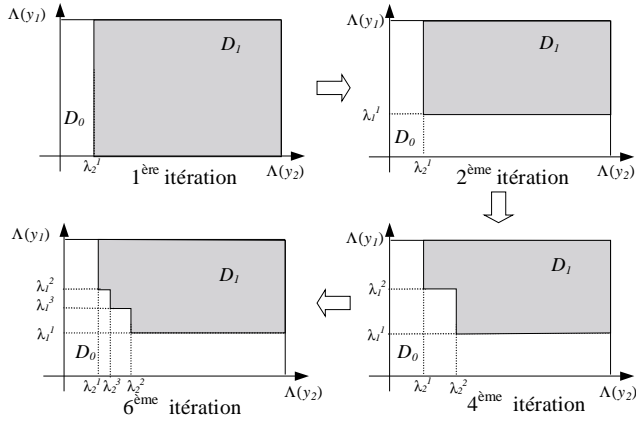


Figure 18. – À chaque itération de l'algorithme, une partition de plus en plus fine de l'espace des observations est créée.

L'algorithme suivant peut être proposé :

```

; DETERMINATION DES PARTITIONS LOCALES
; initialisation
1.  $\forall i = 1, \dots, p$  «  $nb_{\max}(i)$  » et/ou «  $h_{\max}$  » fixés
; boucle
2. pour  $i := 1$  à  $p$  ( $p = \text{nombre de capteurs pertinents}$ )
     $nb(i) \leftarrow 0$ 
     $\lambda_i^{nb(i)} \leftarrow -\infty$ 
    fin pour
; boucle
3. Répéter
4.    pour  $i := 1$  à  $p$ 
5.        Déterminer toutes les dichotomies de  $\Lambda(y_i)$  possibles et...
            ...Calculer  $h(H/\Pi(nb(1), nb(2), \dots, nb(i) + 1, \dots, nb(p)))$ 
        fin pour
6.        Retenir  $y_i$  et mémoriser  $\lambda_i^{nb(i)+1}$  qui minimisent...
            ...  $h(H/\Pi(nb(1), nb(2), \dots, nb(i) + 1, \dots, nb(p)))$ 
7.        si  $nb(i) + 1 > nb_{\max}(i)$  alors Rejeter la dichotomie
            sinon  $nb(i) \leftarrow nb(i) + 1$ 
        fin si
        Jusqu'à  $\{\forall i = 1, \dots, p \quad nb(i) = nb_{\max}(i)\}$  ou
         $\{h(H/\Pi(nb(1), nb(2), \dots, nb(i) + 1, \dots, nb(p))) < h_{\max}\}$ 
; Fin de l'algorithme
    
```

où  $\Pi(nb(1), nb(2), \dots, nb(i) + 1, \dots, nb(p)) = \cap_{i=1}^p \Pi(nb(i))$  et  $\Pi(nb(i))$  représente la partition du domaine d'observations engendrée par le seuillage en  $(i + 1)$  seuils de  $\Lambda(y_i)$ .

## 5.2. construction de l'arbre de décision

Les seuils ayant été déterminés au moyen de l'algorithme précédent, la construction d'un arbre de décision peut être entreprise (figure 19) en minimisant la probabilité d'erreur du système de détection. Chaque nœud de l'arbre consiste à seuiller le rapport de vraisemblance  $\Lambda(y_i)$  ( $i = 1, \dots, p$ ) en  $nb_{\max}(i) + 1$  classes donnant naissance à  $nb_{\max}(i) + 1$  branches, une pour chaque valeur prise par le détecteur local ( $u_i = 0$  ou 1 ou  $(nb_{\max}(i) + 1)$ ).

Chaque feuille permet de prendre une décision finale  $u_0$  ( $H_0$  ou  $H_1$ ), déterminée en minimisant la probabilité d'erreur du processus de détection, ce qui revient à affecter à chaque feuille une décision telle que :  $p(\text{chemin et } H_1) \underset{u_0=H_0}{\overset{u_0=H_1}{\geq}} p(\text{chemin et } H_0)$ .

En reprenant l'exemple du paragraphe précédent, nous nous sommes limités à trois seuils pour chaque capteur, ce qui permet de coder les informations transmises par les quantificateurs locaux à l'opérateur de fusion sur deux bits. L'algorithme proposé détermine les six seuils  $\lambda_{11}, \lambda_{12}, \lambda_{13}, \lambda_{21}, \lambda_{22}$  et  $\lambda_{23}$  en minimisant  $h(H/u_1, u_2)$ .

Avec

$$\begin{aligned}
 u_1 = 0 & \text{ si } \Lambda(y_1) < \lambda_{11} & \text{et} & & u_2 = 0 & \text{ si } \Lambda(y_2) < \lambda_{21} \\
 u_1 = 1 & \text{ si } \lambda_{11} < \Lambda(y_1) < \lambda_{12} & & & u_2 = 1 & \text{ si } \lambda_{21} < \Lambda(y_2) < \lambda_{22} \\
 u_1 = 2 & \text{ si } \lambda_{12} < \Lambda(y_1) < \lambda_{13} & & & u_2 = 2 & \text{ si } \lambda_{22} < \Lambda(y_2) < \lambda_{23} \\
 u_1 = 3 & \text{ si } \lambda_{13} < \Lambda(y_1) & & & u_2 = 3 & \text{ si } \lambda_{23} < \Lambda(y_2)
 \end{aligned}$$

Les valeurs des seuils  $\lambda_{11}, \lambda_{12}, \lambda_{13}$  (resp.  $\lambda_{21}, \lambda_{22}, \lambda_{23}$ ) associées au capteur  $y_1$  (resp.  $y_2$ ) sont indiquées sur la figure 20 (resp. figure 21). La figure 22 représente, quant à elle, la variation de la probabilité d'erreur en fonction de  $P_0$  correspondant à l'arbre de décision construit en considérant :

- les deux capteurs pris séparément,
- la solution distribuée avec deux (paragraphe précédent) et six seuils,
- ainsi que la solution centralisée optimale.

Cette dernière figure met en évidence le fait que la probabilité d'erreur trouvée en considérant six seuils se trouve bien en deçà

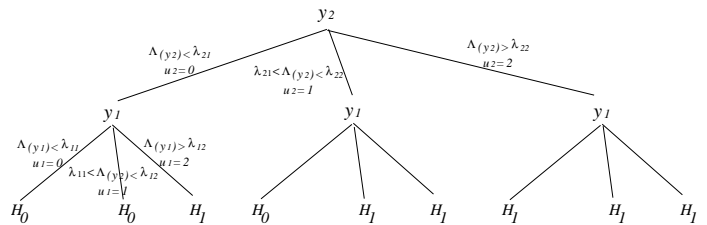
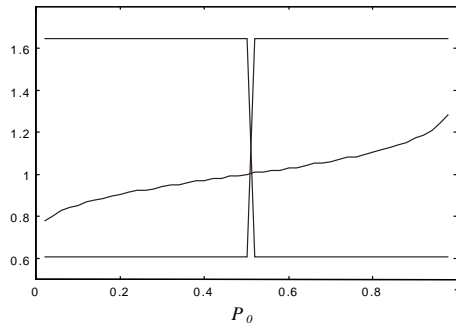
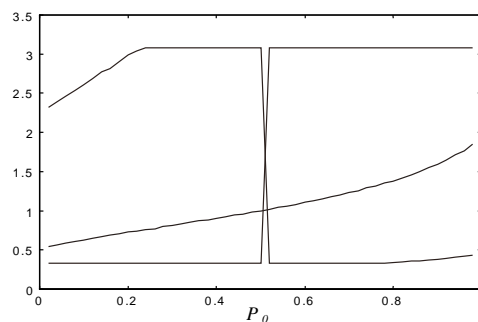
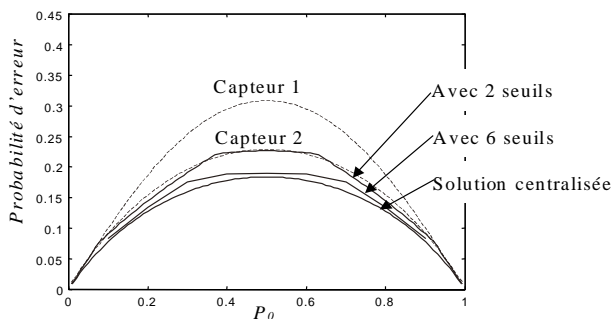


Figure 19. – Exemple d'arbre de décision construit en se limitant à 2 seuils par variable.

Seuils associés à  $y_1$ 

Figure 20. – Seuils associés à  $y_1$  en fonction de  $P_0$ .

Seuils associés à  $y_2$ 

Figure 21. – Seuils associés à  $y_2$  en fonction de  $P_0$ .

Figure 22. – Probabilité d'erreur en fonction de  $P_0$ .

de celle trouvée en ne considérant que deux seuils, quelle que soit la valeur de  $P_0$ . De plus, elle est très proche de la probabilité d'erreur trouvée en considérant la solution optimale centralisée.

## 6. conclusions et perspectives

Le principal avantage des systèmes de détection distribuée par rapport aux systèmes de détection centralisée est de réduire les coûts de communication. Nous avons proposé d'utiliser un

critère basé sur l'entropie afin d'optimiser ces systèmes. Les équations obtenues sont similaires aux équations obtenues avec les critères de Bayes et de Neyman-Pearson. Les coûts ne sont pas des constantes mais dépendent des probabilités *a posteriori*. De façon générale, le nombre d'équations à résoudre simultanément croît très rapidement avec le nombre de capteurs, les calculs nécessaires à la résolution de ces équations deviennent alors très vite inextricables. Dans ce contexte, une phase de sélection de capteurs pertinents pour le processus de décision a été mise en place, et une méthode de détermination des seuils simple à implémenter et peu coûteuse en termes de calculs a été proposée. Afin de trouver un compromis « performances-coûts », les techniques d'optimisation précédentes ont été étendues au problème de la quantification répartie.

Chaque résultat théorique est illustré par un exemple simple couramment étudié dans la littérature. Nous désirons les appliquer à des systèmes plus complexes dans le contexte de la surveillance de processus industriels.

Dans ce cadre, l'indépendance statistique des observations sur laquelle s'appuie toute la démarche de la théorie de la détection semble être une hypothèse trop forte. En utilisant les outils de la théorie de l'information, cette contrainte ne semble pas nécessaire, et une validation théorique devra être établie.

Enfin, il semble indispensable d'étendre les résultats obtenus à des problèmes de décisions multi-hypothèses.

## BIBLIOGRAPHIE

- [1] K.J. Hintz, « A Measure of the Information Gain Attributable to Cueing », IEEE Transactions on Systems, Man, and Cybernetics, vol. 21, pp. 434-442, March/April 1991.
- [2] K. Kastella, « Event-Averaged Maximum Likelihood Estimation and Mean-Field Theory in Multitarget Tracking », IEEE Transactions on Automatic Control, vol. 40, no. 6, pp. 1070-1074, June 1995.
- [3] A. Wald, « Sequential Analysis », Joh Wiley & Sons, New-York, 1947.
- [4] P.M. Frank « General Solution of the Robustness Problem in Linear Fault Detection Filters », European Control Conference 91, pp. 1407-1412, édition Hermès, Grenoble, France, July 2-5, 1991..
- [5] R.R. Teney and N.R. Sandell, « Detection with Distributed Sensors », IEEE Trans. on Aerospace and Electronic Systems, vol. AES-17, n° 4, pp. 98-101, July 1981.
- [6] F.A. Sadjadi, « Hypotheses Testing in a Distributed Environment », IEEE Trans. on Aerospace Electronics Systems, vol. AES, n° 2, pp. 134-137, March 1986.
- [7] P.F. Swaszek, « On the Performance of Serial Networks in Distributed Detection », IEEE Trans. on Aerospace Electronics Systems, vol. 29, n° 1 pp. 254-260, Jan. 1993.
- [8] Z.B. Tang, K.R. Pattipati, and D.L. Kleinman, « A Algorithm for Determining the Decision Thresholds in a Distributed Detection Problem », IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-21, pp. 231-237, Jan-Fev. 19912.
- [9] Z.B. Tang, K.R. Pattipati, and D.L. Kleinman, « Optimisation of Detection Networks : Part I-Tandem Structures », IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-21, n° 5, pp. 1044-1059, Sept-Oct. 1991.
- [10] Z.B. Tang, « Optimisation of Detction Networks », Ph.D. Dissertation, University of Connecticut, 1990.

- [11] P.K. Varshney, « *Distributed Detection and Data Fusion* », Springer Verlag, New-York, 1997.
- [12] Z. Chair and P.K. Varshney, « *Optimal Data Fusion in Multiple Sensor Decision Systems* », IEEE Trans. on Aerospace and Electronic Systems, vol. AES-22, n°1, pp. 98-101, 1986.
- [13] A.R. Reibman and L.W. Nolte, « *Optimal Detection and Performance of Distributed Sensor System* », IEEE Trans. on Aerospace and Electronic Systems, vol. AES-23, n°1, pp. 24-30, Jan. 1987.
- [14] I.Y. Hoballah and P.K. Varshney, « *Distributed Bayesian Signal Detection* », IEEE T. on Information Theory, vol. IT-35, n° 5, pp. 995-1000, Sept. 1989.
- [15] R. Srinivan, « *A Theory of Distributed Detection* », Signal Processing, 11, pp. 319-327, 1986.
- [16] R. Srinivasan, « *Distributed Radar Detection Theory* », IEEE Proc., vol. 133, part F, pp. 55-60, Inst of Electrical Engineers, England, UK, Feb. 1983.
- [17] S.C.A. Thomopoulos, R. Viswanathan, and D.K. Bougoulas, « *Optimal Decision Fusion in Multiple Sensor Systems* », IEEE Trans. on Aerospace and Electronic Systems, vol. AES-23, n°5, pp. 644-653, Sept. 1987.
- [18] S.C.A. Thomopoulos, R. Viswanathan, and D.K. Bougoulas, « *Optimal Distributed Decision Fusion* », IEEE Trans. on Aerospace and Electronic Systems, vol. AES-25, n°5, pp. 761-765, Sept. 1989.
- [19] E. Drakopoulos and C.C. Lee, « *Optimum Multisensor Fusion of Correlated Local Decisions* », IEEE Trans. on Aerospace and Electronic Systems, vol. AES-27, n°4, pp. 593-605, July. 1991.
- [20] M.P. Carton, D. Pomorski, M. Staroswiecki, « *Sélection des capteurs pour systèmes de détection décentralisée par un algorithme d'apprentissage basé sur l'entropie* », Quinzième Colloque sur le Traitement du Signal et des Images (GRETSI), tome 2, pp. 917-920, 18-21 septembre 1995, Juan-les-Pins, France.
- [21] D. Middleton, « *Statistical Communication Theory* », McGraw-Hill, New-York, 1960.
- [22] T.L. Gabrielle, « *Information Criteria for Thresold Determination* », IEEE T. on Information Theory, vol. 6, pp. 484-486, Oct. 1966.
- [23] I.Y. Hoballah and P.K. Varshney, « *An Information Theoretic Approach to the Distributed Detection Problem* », IEEE T. on Information Theory, vol. IT-35, n°5, pp. 988-994, Sept. 1989.
- [24] D. Warren, P. Willet, and R. Rampertab, « *Shannon's Information in Decetralized Signal Detection* », Proc. 23 rd Conf. on Info. Sciences and Systems, Baltimore , Maryland, March 1989.
- [25] C. Desrousseaux, « *Utilisation d'un critère entropique dans les systèmes de détection* », thèse de l'université Lille I, décembre 1998, numéro d'ordre :2381.
- [26] D.A. Huffman, « *A Method for the Construction of Minimum-Redundancy Codes* », in W. Jackson ed., *Communication Theory*, pp. 102-110, Butterworths Scientific Publications, London, 1953.
- [27] J.R. Quinlan, « *Learning Efficient Classification Procedures an their Application to Chess and Games* », in R.S. Michalski, J.G. Carbonell, and T.M. Mitchell editors, *Machine Learning : an Artificial Intelligence Approach*, pp. 463-482, Tioga Publishing Company, 1983.
- [28] C. Desrousseaux, D. Pomorski, « *Optimisation entropiques des systèmes de détection distribuée* », Traitement du Signal, volume 16, n°4, pp. 275-293, septembre 1999.
- [29] V.M. Toro Cordoba, « *Contribution à l'analyse structurale des systèmes complexes à l'aide de l'entropie et ses généralisations* », thèse de l'université de Lille I, mars 1982.

*Manuscrit reçu le 19 mai 1999.*

## LES AUTEURS

Denis POMORSKI



Denis Pomorski est né en 1966. Docteur de l'université Lille I en automatique et informatique industrielle, il est actuellement maître de conférences H.D.R. à l'IUT A de Lille. Son activité de recherche porte essentiellement sur l'utilisation de la théorie de l'information dans le cadre de l'aide au diagnostic et l'optimisation des systèmes de détection.

Christophe DESROUSSEAUX



Christophe Desrousseaux est né en 1971. Ingénieur ISEN, il a soutenu sa thèse de Doctorat en décembre 1998. Son activité de recherche s'est développée autour de la fusion de données et plus particulièrement sur les problèmes d'optimisation des architectures de détection.